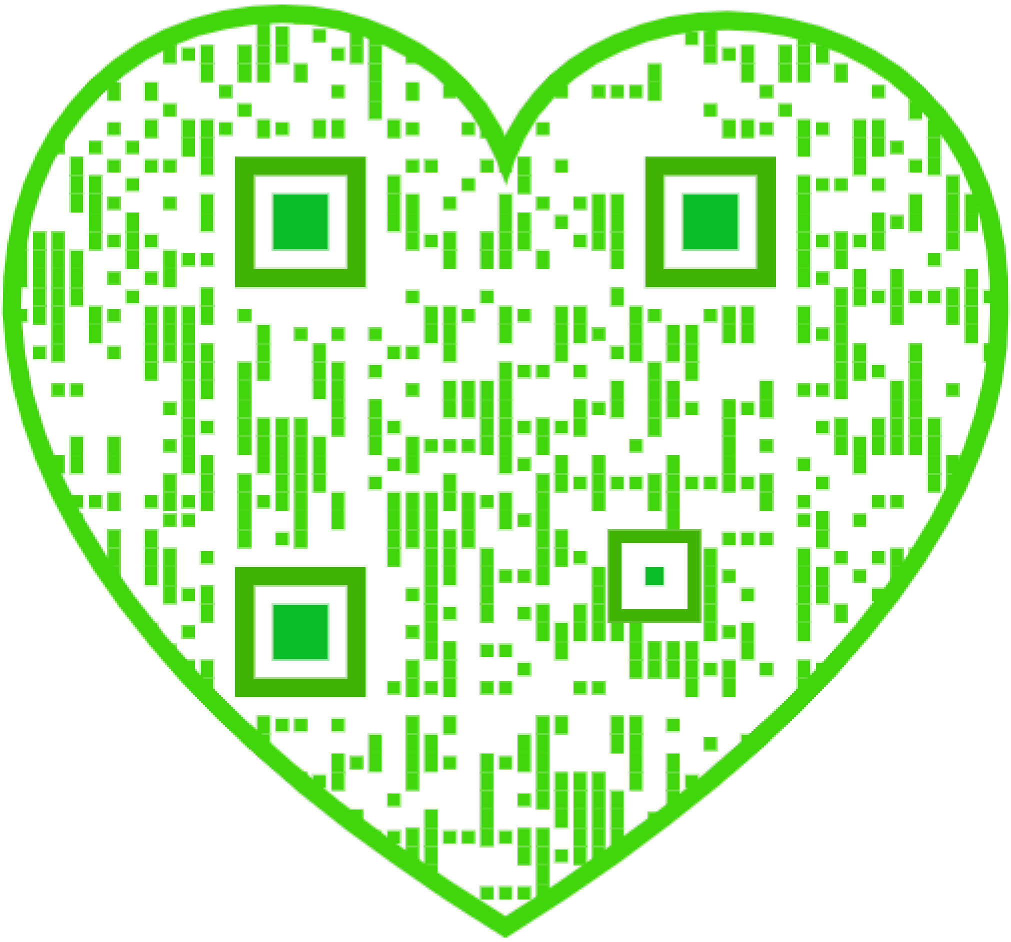
Master in Artificial Intelligence

Feature Engineering IV



Purpose

The purpose of the section is to help you learn how to identify and extract meaningful features from the data to become a Successful Artificial Intelligence (AI) Engineer

At the end of this lecture, you will learn the following

How to Reduce the dimensionality of the feature space while preserving as much relevant information as possible

How to Reduce the dimensionality of the feature space

Understand the Problem and Data

Perform Exploratory Data Analysis (EDA)

Feature Engineering

Feature Selection

Dimensionality Reduction

Evaluate Feature Importance

Iterate and Experiment

Domain Knowledge Integration

Validate and Test

How to reduce the dimensionality of the feature space

Principal Component Analysis (PCA)

t-Distributed Stochastic Neighbor Embedding (t-SNE)

Singular Value Decomposition (SVD)

Principal Component Analysis (PCA)

PCA is a linear dimensionality reduction technique that aims to find the orthogonal axes (principal components) in the feature space that capture the maximum variance in the data

```
from sklearn.decomposition import PCA

# Assuming X contains your features
# Initialize PCA with the desired number of components
n_components = 2  # Number of components to retain
pca = PCA(n_components=n_components)

# Fit PCA to the data and transform it
X_pca = pca.fit_transform(X)
```

t-Distributed Stochastic Neighbor Embedding (t-SNE)

t-SNE is a non-linear dimensionality reduction technique that aims to map high-dimensional data to a lower-dimensional space while preserving local similarities between data points

```
from sklearn.manifold import TSNE

# Assuming X contains your features
# Initialize t-SNE with the desired number of components
n_components = 2  # Number of components in the lower-dimensional space
tsne = TSNE(n_components=n_components)

# Fit t-SNE to the data and transform it
X_tsne = tsne.fit_transform(X)
```

Singular Value Decomposition (SVD)

SVD is a matrix factorization technique that decomposes a matrix into three matrices, allowing for dimensionality reduction by selecting a subset of the components

```
from sklearn.decomposition import TruncatedSVD

# Assuming X contains your features
# Initialize TruncatedSVD with the desired number of
components
n_components = 2  # Number of components to retain
svd = TruncatedSVD(n_components=n_components)

# Fit SVD to the data and transform it
X_svd = svd.fit_transform(X)
```

Selecting the Number of Components

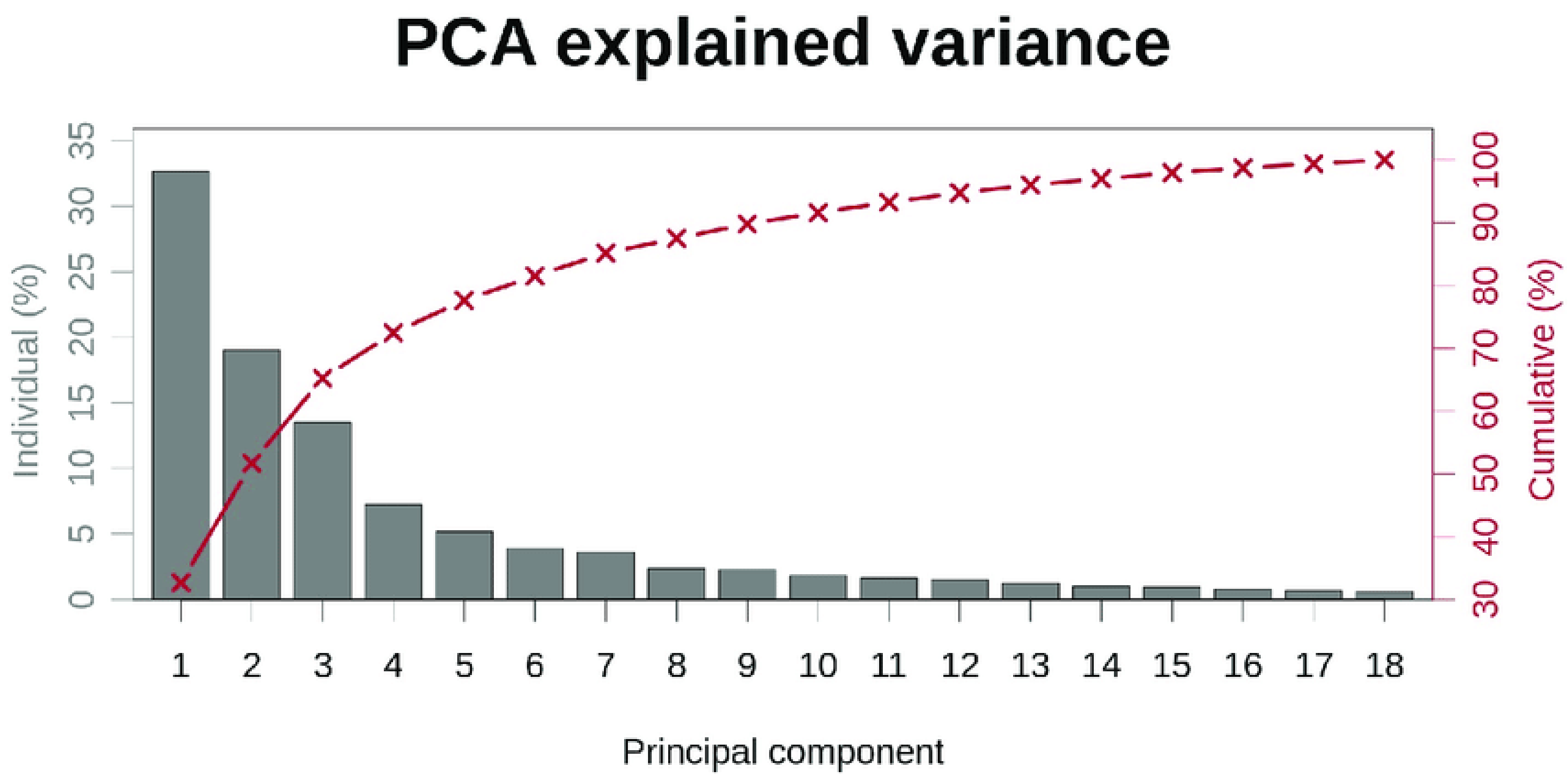
PCA and SVD

- Explained variance ratio
- Cumulative explained variance

t-SNE

- Specified beforehand
- Based on the desired dimensionality of the lower-dimensional space

What is Explained Variance Technique?



Visualization

```
import matplotlib.pyplot as plt

# Assuming y contains your target variable or cluster labels
plt.scatter(X_pca[:, 0], X_pca[:, 1], c=y)
plt.title('PCA Visualization')
plt.xlabel('Principal Component 1')
plt.ylabel('Principal Component 2')
plt.show()
```

Modeling with Reduced Dimensionality

Reduced-dimensional data

- Downstream tasks such a
  - Classification
  - Clustering
  - Visualization
- Train machine learning models or apply clustering algorithms
  - Perform tasks on the reduced-dimensional space

How to reduce the dimensionality of the feature space

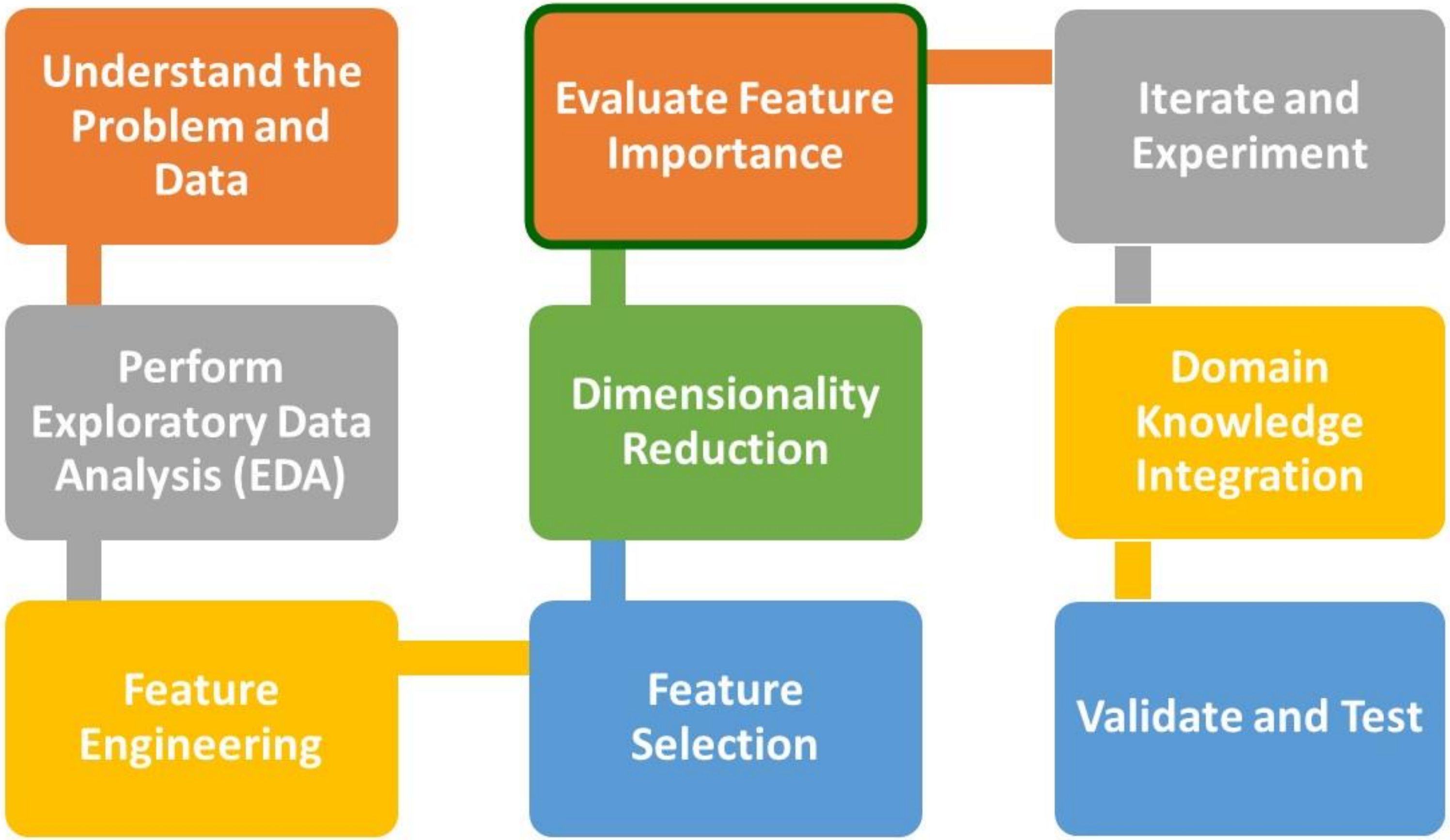
Principal Component Analysis (PCA)

t-Distributed Stochastic Neighbor Embedding (t-SNE)

Singular Value Decomposition (SVD)

What is next?

Then, Evaluate Feature Importance



Master in Artificial Intelligence

Feature Engineering IV